

**REMARKS**

In view of the following remarks, Applicant respectfully requests reconsideration and allowance of the subject application.

**§103 Rejections**

Claims 1-26, 33 and 37 are rejected under 35 U.S.C. §103(a) as allegedly being unpatentable over US Patent No. 6,560,591 to Memmott et al. (hereinafter, "Memmott") in view of Admitted Prior (hereinafter, "APA") and further in view of "Using the Command-Line Utility" (hereinafter, "UT"). Applicant respectfully traverses the rejection.

Memmott teaches a system for managing data providers. A data requester 110 forwards a query to a data resolver 120, which chooses a priority list of data providers 130 from a set of lists based on the characteristic of the query. The data resolver 120 forwards a request to a data provider 130 in the list based on the query. The data resolver 120 receives data in response to the request and returns a response to the data requestor 110 based on the data. (col. 3, ln 7-62; col. 4, ln 1-21).

Applicant's claim 1 recites:

A command line utility embodied in one or more computer-readable media, the command line utility comprising:  
an object model command schema to define a mapping between one or more commands and an object model target schema, the one or more commands generated by the command schema and configured to operate against the target schema through the command line utility.

Regarding claim 1, the Office asserts, at Page 2 of the Office Action, that Memmott teaches the invention substantially as claimed including: an "X utility",

1 an object model command schema, a mapping, one or more commands, and an  
2 object target schema. The Office asserts that Memmott's data resolver 120, data  
3 provider 130, selection task P120, and interface module 140, comprise an "X  
4 utility". However, these elements do not teach or suggest a "command line  
5 utility" as recited in applicant's claim 1. Memmott's data resolver 120 is an  
6 application that executes on a client to receive a query and choose a priority list  
7 from a list of data providers 130 based on the query characteristic. In one  
8 embodiment of Memmott, a data resolver may communicate with a data provider  
9 through an interface module 140. These elements of Memmott do not teach or  
10 suggest a "command line utility" as recited in applicant's claim 1.

11 The Office further asserts that Memmott teaches "an object model  
12 command schema" at col. 8, ln 41-55 and col. 7, ln 40-66. However, in col. 7,  
13 Memmott simply mentions that disk space is reported in units of bytes in the CIM  
14 distributed management scheme, while disk space is reported in units of kilobytes  
15 in the DMI scheme. This mere mention of the CIM distributed management  
16 scheme in Memmott cannot fairly be said to be "an object model command  
17 schema" as recited in Applicant's claim 1. At col. 8, Memmott further gives an  
18 example of a data provider which uses the CIM distributed management scheme as  
19 being the preferred data provider. This additional mention of the CIM distributed  
20 management scheme in Memmott cannot fairly be considered "an object model  
21 command schema". Nowhere, in fact, does Memmott teach or suggest "an object  
22 model command schema" as the Office asserts. Although CIM is a method for  
23 describing management information that relies on inheritance and other object-  
24 oriented features, Memmott's mere mention or use of CIM distributed  
25 management scheme in no way teaches or implies "an object model command

1 schema" as recited in Applicant's claim 1. Memmott does not teach or discuss  
2 any sort of "command schema", let alone "an object model command schema".

3 Furthermore, Applicant's claim 1 recites "an object model command  
4 schema to define a mapping between one or more commands and an object model  
5 target schema". The Office has not pointed to anything in Memmott or any other  
6 reference that teaches or suggests "an object model command schema to define a  
7 mapping". Memmott does not teach or suggest "an object model command  
8 schema to define a mapping", or that such a mapping is a "mapping between one  
9 or more commands and an object model target schema" as recited in Applicant's  
10 claim 1.

11 The Office asserts that Memmott teaches "a mapping" at col. 5, ln 47-60.  
12 However, as noted, Applicant's claim 1 recites "a mapping between one or more  
13 commands and an object model target schema". Memmott teaches a string that  
14 represents a mapping between a portion of a query and a data provider identifier.  
15 This is not the same as "a mapping between one or more commands and an object  
16 model target schema". Furthermore, as mentioned above, Memmott does not  
17 teach or suggest "an object model command schema to define a mapping" as  
18 recited in Applicant's claim 1. Because Memmott does not teach "an object model  
19 command schema" at all, it cannot fairly be said that Memmott teaches "an object  
20 model command schema to define a mapping". Thus, it also cannot fairly be said  
21 that Memmott teaches "an object model command schema to define a mapping  
22 between one or more commands and an object model target schema", as recited in  
23 Applicant's claim 1.

24 At Page 3 of the Office Action, the Office admits that Memmott does not  
25 teach that commands are generated by the command schema. However, the Office

1 asserts that this is taught by Applicant's own APA (admitted prior art).  
2 Specifically, the Office points to page 3, ln 7-18, of APA and asserts that the  
3 phrase "made up of classes representing management applications" is the same as  
4 the claimed "one or more commands generated by the command schema", and that  
5 it would have been obvious to combine Memmott and APA because "APA's  
6 'made up of classes' would provides [sic] access to management information on a  
7 single network machine, or a large number of machines all at once". Applicant  
8 first notes that at page 3, ln 7-18, there is merely a brief discussion of CIM  
9 schema. The APA mentions that companies can create extension schemas that are  
10 made up of classes representing managed objects. Creating extension schemas  
11 that are made up of classes representing managed objects is not the same as  
12 Applicant's claimed "one or more commands generated by the command schema",  
13 as the Office asserts. There is, in fact, no mention at all in APA about "one or  
14 more commands generated by the command schema".

15 A prima facie case of obviousness requires that the prior art reference (or  
16 references when combined) must teach or suggest all the claim limitations (MPEP  
17 2142, 2143). However, it is clear from the above discussion, that various elements  
18 recited in Applicant's claim 1 are not taught or suggested by Memmott and APA,  
19 alone or in combination. Furthermore, the various elements discussed above and  
20 recited in Applicant's claim 1 are not taught or suggested by any other references  
21 relied upon by the Office. For at least the numerous reasons above showing that  
22 Memmott and APA, alone or in combination, fail to teach or suggest all the claim  
23 limitations of claim 1, a prima facie case of obviousness is not supported.  
24 Applicant therefore respectfully requests that the §103(a) rejection to claim 1 be  
25 removed.

1 The Office next refers to UT ("Using the Command-Line Utility") for  
2 support. The Office cites UT only for its purported discussion of a "command  
3 line", and not for any suggestion of the various elements of Applicant's claim 1  
4 discussed above. Accordingly, UT does not remedy the deficiencies of Memmott  
5 and APA noted above, and claim 1 is allowable over the combination of these  
6 three references.

7 Furthermore, regarding claim 1, the Office admits that Memmott and APA  
8 do not teach "X" as a command-line utility. The Office asserts, however, that UT  
9 teaches a command line at page 1, and that it would have been obvious to combine  
10 Memmott, APA and UT, "because UT's Command line would performs [sic] a  
11 transformation". Applicant notes, however, that UT merely discusses a command-  
12 line interface that performs an XSL transformation. UT does not teach anything  
13 about a command line utility as recited in Applicant's claim 1 which comprises:

14 an object model command schema to define a mapping  
15 between one or more commands and an object model target schema,  
16 the one or more commands generated by the command schema and  
17 configured to operate against the target schema through the command  
line utility.

18 Furthermore, in addition to requiring that the prior art reference (or  
19 references when combined) teach or suggest all the claim limitations, a prima facie  
20 case of obviousness requires that there be some suggestion or motivation to  
21 modify a reference or to combine reference teachings (MPEP 2143). The teaching  
22 or suggestion to make the claimed combination must be found in the prior art, not  
23 in applicant's disclosure (MPEP 2143; In re Vaack, 947 F.2d 488, 20 USPQ2d  
24 1438 (Fed. Cir. 1991)). The mere fact that references can be combined or  
25 modified does not render the resultant combination obvious unless the prior art

1 also suggests the desirability of the combination (MPEP 2143.01; *In re Mills*, 916  
2 F.2d 680, 16 USPQ2d 1430 (Fed. Cir. 1990)).

3 Applicant respectfully submits that there is no suggestion or motivation to  
4 combine the teachings of Memmott and UT. As noted above, Memmott teaches a  
5 system for managing data providers, while UT teaches a command-line interface  
6 that performs an XSL transformation. But for the suggestion in Applicant's claim  
7 1 of a command line utility, there would be no reason for one skilled in the art to  
8 combine the teachings of Memmott and UT. Applicant respectfully submits that  
9 such a combination of the teachings of Memmott and UT can only be made  
10 through the use of impermissible hindsight reconstruction with Applicant's claim  
11 as a guide. Such a combination is therefore improper.

12 For the additional reasons that UT does not teach anything about a  
13 command line utility as recited in Applicant's claim 1, and that the combination of  
14 the Memmott and UT references is improper and based on hindsight  
15 reconstruction, Applicant respectfully submits that a prima facie case of  
16 obviousness is not supported with regard to Applicant's claim 1. Applicant  
17 therefore respectfully requests that the §103(a) rejection to claim 1 be removed.

18 Claims 2-15 depend from claim 1 and therefore include the elements of  
19 claim 1. Therefore, claims 2-15 are allowable at least on the basis of this  
20 dependency, in addition to the further elements recited therein which are neither  
21 shown nor suggested by the cited references. Accordingly, Applicant respectfully  
22 requests that the 35 U.S.C. §103(a) rejection to claims 2-15 be removed.

23 Independent claim 16 recites:

24 An object model schema embodied in one or more computer-  
25 readable media, the object model schema comprising:

1 an alias class to define alias instances, each alias  
instance representing a command;

2 a verb class to define verb instances, each verb instance  
representing behavior available through an alias instance;

3 a parameter class to define parameters accepted by a  
verb instance;

4 a format class to define format instances, each format  
instance having a list of properties to be displayed through an alias  
instance;

6 a property class to define property instances, each  
property instance representing a property value from a property list;

7 a connection class to define connection instances, each  
connection instance representing connection parameters used by an  
alias instance to establish a connection to the target schema;

9 a qualifier class to define qualifier instances, each  
qualifier instance representing constraints on elements of an alias  
instance;

10 a localized string class to define localized string  
instances, each localized string instance representing a text  
localization for translating text into a localized language; and

12 a see-also association to associate an alias instance with  
other related alias instances.

14 On page 5 of the current Office Action, the Office rejects independent  
15 claim 16 for the same reasons it rejects claim 12. Furthermore, the Office rejects  
16 claim 12 for the same reasons it rejects claims 2-8. Regarding claim 2, the Office  
17 asserts that Memmott teaches an alias class at col. 5, ln 18-30 and col. 4, ln 40-60.  
18 However, Memmott merely discusses query characteristics that indicate different  
19 classes, e.g., class 1 and class 2. Nowhere does Memmott teach or suggest "an  
20 alias class to define alias instances, each alias instance representing a command"  
21 as recited in Applicant's claim 16. Thus, the rejection of claim 16 is not supported  
22 and should be removed.

23 Further regarding the rejection of claim 16, the Office asserts that with  
24 respect to claim 3, Memmott teaches a verb class, a format class, and a connection  
25

1 class as a subclass at col. 5, ln 17-30 and col. 4 ln 40-60. However, the words  
2 "verb class", "format class", and "connection class" do not appear in any form  
3 throughout the entire text of Memmott. Furthermore, there is no discussion  
4 whatsoever in Memmott that teaches, suggests, or implies a "verb class", "format  
5 class", or a "connection class". As shown above, Applicant's claim 16 recites,

6 a verb class to define verb instances, each verb instance  
7 representing behavior available through an alias instance;  
8 a format class to define format instances, each format instance  
9 having a list of properties to be displayed through an alias instance;  
10 a connection class to define connection instances, each  
11 connection instance representing connection parameters used by an  
12 alias instance to establish a connection to the target schema;

13 For the additional reasons that Memmott does not teach or suggest a "verb  
14 class to define verb instances, each . . .", "format class to define format instances,  
15 each . . .", or a "connection class to define connection instances, each . . ." as  
16 recited in Applicant's claim 16, the rejection of claim 16 is not supported and  
17 should be removed.

18 Further regarding the rejection of claim 16, the Office asserts that with  
19 respect to claim 4, Memmott teaches at col. 5, ln 18-30, a parameter class as a  
20 subclass with each instance of the parameter class representing parameters:  
21 However, claim 16 recites,

22 a verb class to define verb instances, each verb instance  
23 representing behavior available through an alias instance;  
24 a parameter class to define parameters accepted by a verb  
25 instance;

26 As noted above, Memmott does not teach or suggest "a verb class to define  
27 verb instances". Thus, it cannot fairly be said that Memmott teaches "a parameter



1 class to define parameters accepted by a verb instance". Furthermore, the words  
2 "parameter class" do not appear in any form throughout the entire text of  
3 Memmott. Moreover, there is no discussion in Memmott that teaches, suggests, or  
4 implies a "parameter class to define parameters accepted by a verb instance".  
5 Accordingly, for the additional reason that Memmott does not teach or suggest a  
6 "parameter class" as recited in Applicant's claim 16, the rejection of claim 16 is  
7 not supported and should be removed.

8 Further regarding the rejection of claim 16, the Office asserts that with  
9 respect to claim 5, Memmott teaches at col. 5, ln 18-30, a property class as a  
10 subclass to the format class with each instance of the property class representing a  
11 property value. With respect to a property class, claim 16 recites,

12 a format class to define format instances, each format instance  
13 having a list of properties to be displayed through an alias instance;  
14 a property class to define property instances, each property  
15 instance representing a property value from a property list;

16 First of all, there is no discussion or teaching in Memmott regarding a  
17 "format class to define format instances, each format instance having a list of  
18 properties to be displayed through an alias instance". Thus, it cannot fairly be said  
19 that Memmott teaches "a property class to define property instances, each property  
20 instance representing a property value from a property list". Furthermore, the  
21 words "property class", "property instances", "property list", "format class", etc.,  
22 do not appear in any form throughout the entire text of Memmott. Moreover, there  
23 is no discussion whatsoever in Memmott that teaches, suggests, or implies  
24 anything about a "property class". Accordingly, for these additional reasons, the  
25 rejection of claim 16 is not supported and should be removed.

1       The very same arguments stated above regarding certain elements of claim  
2 16, can be equally applied to the various other elements of claim 16. That is,  
3 Memmott does not teach, suggest, or imply anything regarding elements including  
4 "a qualifier class", "a localized string class", or "a see-also association".

5       For at least all the numerous reasons stated above, the rejection of claim 16  
6 is not supported. Accordingly, Applicant respectfully requests that the rejection to  
7 claim 16 be removed.

8       Regarding independent claim 17, the Office rejects claim 17 for the same  
9 reasons it rejects claim 1. The elements of claim 17 parallel those discussed above  
10 with respect to claim 1. For example, claim 17 recites in part:

11               a set of commands generated by an object model  
12       command schema to operate against an object model target schema,  
13       the command schema defining a mapping between the set of  
14       commands and the target schema; and  
15               an interface utility to facilitate implementation of  
16       individual commands within the set of commands.

17       Therefore, the reasoning stated herein above regarding the rejection of  
18 claim 1 is similarly applicable to the rejection of claim 17. For example, none of  
19 the cited references teaches or suggests "a set of commands generated by an object  
20 model command schema". The Office asserts, with respect to claim 1, that APA  
21 teaches commands generated by the command schema. However, as noted above,  
22 the APA merely provides a brief discussion of CIM schema and mentions that  
23 companies can create extension schemas that are made up of classes representing  
24 managed objects. Creating extension schemas that are made up of classes  
25 representing managed objects is not the same as "commands generated by an  
object model command schema". There is no mention at all in APA about

1 "commands generated by an object model command schema". Accordingly, for  
2 at least the various reasons stated above regarding claim 1, Applicant respectfully  
3 submits that a prima facie case of obviousness is not supported with respect to  
4 claim 17. Applicant therefore respectfully requests that the §103(a) rejection to  
5 claim 17 be removed.

6 Claims 18-23 depend from claim 17 and therefore include the elements of  
7 claim 17. Therefore, claims 18-23 are allowable at least on the basis of this  
8 dependency, in addition to the further elements recited therein which are neither  
9 shown nor suggested by the cited references. Accordingly, Applicant respectfully  
10 requests that the 35 U.S.C. §103(a) rejection to claims 18-23 be removed.

11 Regarding independent claim 24, the Office rejects claim 24 for the same  
12 reasons it rejects claims 1 and 12. Claim 24 recites the following:

13 A management application embodied in one or more computer-  
14 readable media, the management application comprising:

15 a first object model to control the configuration and  
16 behavior of the management application in operating against and  
17 managing a second object model.

18 Although the Office rejects claim 24 for the same reasons it rejects claims 1  
19 and 12, in its rejection of claim 1 and 12, the Office does not point out anything in  
20 any of the cited references that teach or suggest the elements of claim 24.  
21 Furthermore, a thorough review of the cited references reveals that the references  
22 do not teach or suggest the elements of claim 24. Specifically, none of the cited  
23 references teaches or suggests at least "a first object model to control the  
24 configuration and behavior of the management application in operating against  
25 and managing a second object model" as recited in claim 24. Accordingly, the

1 rejection of claim 24 cannot stand, and Applicant respectfully requests that the 35  
2 U.S.C. §103(a) rejection to claim 24 be removed.

3       **Claims 25-26** depend from claim 24 and therefore include the elements of  
4 claim 24. Therefore, claims 25-26 are allowable at least on the basis of this  
5 dependency, in addition to the further elements recited therein which are neither  
6 shown nor suggested by the cited references. Accordingly, Applicant respectfully  
7 requests that the 35 U.S.C. §103(a) rejection to claims 25-26 be removed.

8       Regarding independent claim 33, the Office rejects claim 33 for the same  
9 reasons it rejects claim 1. Claim 33 recites the following:

10       A method of managing objects in a target schema comprising:  
11       providing a user interface;  
12       defining a command structure through an object-oriented  
13       command schema, the command schema including an alias class;  
14       instantiating an object of the alias class as an alias by receiving  
15       parameters of the alias class through the user interface, the alias  
16       representing a command which maps to an object in the target  
17       schema; and  
18       executing the command against the object in the target schema.

19       To the extent elements of claim 33 parallel elements recited in claim 1 (e.g.,  
20 "defining a command structure through an object-oriented command schema, the  
21 command schema including an alias class"), arguments already presented above  
22 regarding the rejection of claim 1 apply similarly to the rejection of claim 33. For  
23 these reasons alone, claim 33 is allowable over the cited references and the  
24 rejection to claim 33 should be removed.

25       Furthermore, there is no teaching or suggestion in the cited references of  
the various additional elements of claim 33, including, "instantiating an object of  
the alias class as an alias by receiving parameters of the alias class through the

1 user interface", or "the alias representing a command which maps to an object in  
2 the target schema". The Office Action does not point to anything in the cited  
3 references which teaches or suggests these elements of claim 33. Further,  
4 Applicant is unable to find any teaching or suggestion of such elements in any of  
5 the cited references. Accordingly, for these additional reasons, claim 33 is  
6 allowable over the cited references and the rejection to claim 33 should be  
7 removed.

8 Claims 34-37 depend from claim 33 and therefore include the elements of  
9 claim 33. Therefore, claims 34-37 are allowable at least on the basis of this  
10 dependency, in addition to the further elements recited therein which are neither  
11 shown nor suggested by the cited references. Accordingly, Applicant respectfully  
12 requests that the 35 U.S.C. §103(a) rejection to claims 34-37 be removed.

13 The Office rejects claims 27-32 under 35 U.S.C. §103(a) as allegedly being  
14 unpatentable over Memmott in view of Scumpu (Session 3: CIM Diagnostics  
15 Development Guidelines and Demo) (hereinafter, "Scumpu") and further in view  
16 of Steve (Network and System Management with XML) (hereinafter, "Steve").  
17 Applicant respectfully traverses the rejection.

18 Claim 27 recites, in part, the following:

19 receiving a command through a command line interface;  
20 fetching an alias for the command;  
21 interpreting the command based on the alias and the current  
22 operating environment of the command line interface;  
23 executing the command as one or more WMI API calls against  
24 a target namespace;  
25 receiving WMI data in XML form;  
applying an XSL style sheet format the WMI data; and  
presenting the WMI data through the command line interface.

1       Regarding claim 27, the Office asserts that Memmott teaches all the  
2 elements of claim 27 except the "WMI API". The Office points to Memmott at  
3 various locations in cols. 3, 4, 5, 6, 8, and 9. With regard to the element of claim  
4 27 which recites "defining a command structure through an object-oriented  
5 command schema, the command schema including an alias class", however,  
6 Applicant has already shown above regarding claim 1, that Memmott does not  
7 teach or suggest that commands or a command structure are generated by an  
8 object-oriented command schema.

9       Furthermore, Memmott does not teach or suggest "fetching an alias for the  
10 command", where the "command" is received "through a command line  
11 interface", as generally recited in claim 27. The Office does not point to anything  
12 in Memmott or any other reference that teaches or suggests such elements. In  
13 addition, Memmott does not teach or suggest "interpreting the command based on  
14 the alias and the current operating environment of the command line interface",  
15 where the "command" is received "through a command line interface" and the  
16 "alias" is fetched "for the command". In addition, Memmott does not teach or  
17 suggest "executing the command as one or more WMI API calls against a target  
18 namespace", where the "command" is received "through a command line  
19 interface". In addition, Memmott does not teach or suggest "receiving WMI data  
20 in XML form", or "applying an XSL style sheet format the WMI data", or  
21 "presenting the WMI data through the command line interface", all as recited in  
22 Applicant's claim 27. The Office has not pointed to anything in Memmott or any  
23 other reference that teaches or suggests these elements as recited in claim 27.

24       The Office is invited to point to specific locations in Memmott or within  
25 any of the other cited references, where such elements of claim 27 are taught,

1 suggested, or implied in any way. Applicant respectfully submits that such  
2 teachings, suggestions, or implications, do not exist in any of the cited references.  
3 Accordingly, the rejection of claim 27 cannot stand, and Applicant respectfully  
4 requests that the rejection of claim 27 be removed.

5 Furthermore, regarding claim 27, the Office admits that Memmott does not  
6 teach "WMI API", and refers to Scumpu (pg 5, ln 7) for support of such teaching.  
7 Scumpu is a "slideshow" that provides a very general outline of CIM, that has  
8 clearly been prepared to guide a speaker through a speech. It is true that the  
9 Scumpu slideshow mentions "WMI API" on pg 5. Scumpu also illustrates "WMI  
10 API" on pg 3, as part of an overall WMI Architecture. However, Scumpu  
11 provides no information regarding "WMI API". In addition, Scumpu provides no  
12 teaching regarding "executing the command as one or more WMI API calls  
13 against a target namespace", where the "command" is received "through a  
14 command line interface" as is recited in Applicant's claim 27. Furthermore,  
15 Scumpu provides no teaching or suggestion regarding the various other elements  
16 of claim 27. For these additional reasons, Applicant respectfully submits that none  
17 of the cited references teach or suggest the elements of claim 27, and that the  
18 rejection of claim 27 therefore should be removed. Applicant respectfully requests  
19 that the rejection of claim 27 be withdrawn.

20 In addition, as noted above, a prima facie case of obviousness requires that  
21 there be some suggestion or motivation to modify a reference or to combine  
22 reference teachings (MPEP 2143). The teaching or suggestion to make the  
23 claimed combination must be found in the prior art, not in applicant's disclosure  
24 (MPEP 2143; In re Vaack, 947 F.2d 488, 20 USPQ2d 1438 (Fed. Cir. 1991)).  
25 The mere fact that references can be combined or modified does not render the

1 resultant combination obvious unless the prior art also suggests the desirability of  
2 the combination (MPEP 2143.01; In re Mills, 916 F.2d 680, 16 USPQ2d 1430  
3 (Fed. Cir. 1990)).

4 Applicant respectfully submits that there is no suggestion or motivation  
5 whatsoever to combine the teachings of Memmott and Scumpu. Scumpu is a  
6 slideshow which briefly covers "CIM Diagnostics Development Guidelines and  
7 Demo". There is nothing in Scumpu to suggest to one skilled in the art that the  
8 teachings of Memmott and Scumpu should be combined. Applicant respectfully  
9 submits that such a combination of the teachings of Memmott and Scumpu can  
10 only be made through the use of impermissible hindsight reconstruction with  
11 Applicant's claim as a guide. Such a combination is therefore improper.

12 For the additional reason that the combination of the Memmott and Scumpu  
13 references is improper and based on hindsight reconstruction, Applicant  
14 respectfully submits that a prima facie case of obviousness is not supported with  
15 regard to Applicant's claim 27. Applicant therefore respectfully requests that the  
16 §103(a) rejection to claim 27 be removed.

17 Furthermore, regarding claim 27, the Office admits that Memmott and  
18 Scumpu do not teach a "Y" command line applying an XSL style sheet. However,  
19 the Office refers to Steve for support and asserts that Steve teaches a command  
20 line and an XSL style sheet. Steve provides a very broad statement regarding  
21 "Network and Systems Management with XML". Included in Steve are general  
22 discussions of CIM and XML. Steve mentions on pg 4, ln 38 - pg 5, ln 8, that "A  
23 forthcoming new standard . . . is the Extensible Style Language (XSL)" and that  
24 "a command-line interface could be displayed as a style-sheet-defined view . . .  
25 expressed with XML". However, claim 27 recites "fetching an alias for the



1 command", where the "command" is received "through a command line  
2 interface". Such elements are not taught or suggested by Steve, and as noted  
3 above, are also not taught or suggested by the other cited references. Claim 27  
4 further recites, "interpreting the command based on the alias and the current  
5 operating environment of the command line interface", where the "command" is  
6 received "through a command line interface" and the "alias" is fetched "for the  
7 command". Claim 27 also recites, "executing the command as one or more WMI  
8 API calls against a target namespace", where the "command" is received "through  
9 a command line interface". For these additional reasons, Applicant respectfully  
10 submits that a prima facie case of obviousness is not supported with regard to  
11 Applicant's claim 27, and respectfully requests that the §103(a) rejection to claim  
12 27 be removed.

13       **Claims 28-32** depend from claim 27 and therefore include the elements of  
14 claim 27. Therefore, claims 28-32 are allowable at least on the basis of this  
15 dependency, in addition to the further elements recited therein which are neither  
16 shown nor suggested by the cited references. Accordingly, Applicant respectfully  
17 requests that the 35 U.S.C. §103(a) rejection to claims 28-32 be removed.

18  
19  
20 **Conclusion**

21       All pending claims are in condition for allowance. Applicant respectfully  
22 requests reconsideration and prompt issuance of the subject application. If any  
23 issues remain that prevent issuance of this application, the Examiner is urged to  
24 contact the undersigned attorney before issuing a subsequent Action.  
25

Respectfully Submitted,

Dated: 6/30/04

By: Nathan R. Rieth

Nathan R. Rieth  
Reg. No. 44302  
(509) 324-9256; X233